

Claims:

1. A method for testing computing devices, comprising the steps of:

5 providing a suite of test programs on a server for execution by a plurality of said computing devices that are coupled to said server;

distributing different ones of said test programs from said server to said computing devices for concurrent execution
10 thereof by said computing devices;

receiving messages from said computing devices upon completion of respective said different ones of said test programs; and

responsively to said messages, iterating said step of
15 distributing until all of said test programs in said suite have been executed.

2. The method according to claim 1, wherein said test programs are distributed as JAR files and JAD files.

20 3. The method according to claim 2, wherein said JAD files are constructed responsively to said messages.

4. The method according to claim 1, further comprising the
25 steps of:

dynamically coupling a new computing device to said server;
and

reallocating said test programs to said computing devices
and said new computing device.

30 5. The method according to claim 1, further comprising the steps of:

dynamically detaching one of said computing devices from said server; and

marking unexecuted ones of said test programs that were distributed to said one computing device as not run.

5

6. The method according to claim 1, wherein said step of distributing comprises removing said different ones of said test programs from a stack.

10 7. The method according to claim 1, wherein said step of distributing comprises assigning said different ones of said test programs in groups comprising a plurality of said test programs so as to minimize a completion time of said suite.

15 8. A computer software product, comprising a computer-readable medium in which computer program instructions are stored, which instructions, when read by a computer, cause the computer to perform a method for testing computing devices, comprising the steps of:

20 accessing a suite of test programs on a server for execution by a plurality of said computing devices that are coupled to said server;

distributing different ones of said test programs from said server to said computing devices for concurrent execution
25 thereof by said computing devices;

receiving messages from said computing devices upon completion of respective said different ones of said test programs; and

30 responsively to said messages, iterating said step of distributing until all of said test programs in said suite have been executed.

9. The computer software product according to claim 8, wherein said test programs are distributed as JAR files and JAD files.

5 10. The computer software product according to claim 9, wherein said computer is further instructed to construct said JAD files responsively to said messages.

10 11. The computer software product according to claim 8, wherein said computer is further instructed to perform the steps of:

 dynamically coupling a new computing device to said server;
 and

 reallocating said test programs to said computing devices
15 and said new computing device.

 12. The computer software product according to claim 8, wherein said computer is further instructed to perform the steps of:

20 dynamically detaching one of said computing devices from said server; and

 marking unexecuted ones of said test programs that were distributed to said one computing device as not run.

25 13. The computer software product according to claim 8, wherein said step of distributing comprises removing said different ones of said test programs from a stack.

30 14. The computer software product according to claim 8, wherein said step of distributing comprises assigning said different ones of said test programs in groups comprising a

plurality of said test programs so as to minimize a completion time of said suite.

15 steps of:

providing a suite of test programs on a server for execution by a plurality of said computing devices that are coupled to said server;

10 assigning a respective unique identifier to each of said plurality of said computing devices, for use in communicating with said server;

making respective allocations comprising different ones of said test programs for said computing devices;

15 downloading said allocations from said server for respective execution by said computing devices coupled thereto, so that at least first and second computing devices among said plurality execute different first and second test programs from said suite substantially simultaneously;

20 receiving messages at said server from said computing devices with respect to said execution of said test programs, each of said messages containing said respective unique identifier; and

25 responsively to each of said messages, downloading at least another of said test programs to a respective one of said computing devices.

16. The method according to claim 15, wherein said step of making respective allocations is performed so as to minimize a completion time of said suite of test programs.

30 17. The method according to claim 15, further comprising the steps of:

coupling a new computing device to said server; and
reallocating said test programs to said computing devices
and said new computing device.

5 18. The method according to claim 15, further comprising
the steps of:

detaching an attached one of said computing devices from
said server; and

10 marking unexecuted tests of said respective allocations of
said attached one computing device as not run.

19. The method according to claim 15, wherein said
computing devices comprise MIDP-compliant devices, and

15 wherein said test programs comprise MIDlets, which are
packaged in respective JAD files and JAR files, and

wherein allocating said test programs comprises downloading
said JAD files and said JAR files to said MIDP-compliant
devices.

20 20. A computer software product, comprising a computer-
readable medium in which computer program instructions are
stored, which instructions, when read by a computer, cause the
computer to perform a method for testing computing devices,
comprising the steps of:

25 accessing a suite of test programs that are stored on a
server for execution by a plurality of said computing devices
that are coupled to said server;

30 assigning a respective unique identifier to each of said
plurality of said computing devices, for use in communicating
with said server;

making respective allocations comprising different ones of
said test programs for said computing devices;

downloading said allocations from said server for
respective execution by said computing devices coupled thereto,
so that at least first and second computing devices among said
plurality execute different first and second test programs from
5 said suite substantially simultaneously;

receiving messages at said server from said computing
devices with respect to said execution of said test programs,
each of said messages containing said respective unique
identifier; and

10 responsively to each of said messages, returning a new
allocation of unexecuted ones of said test programs to
respective ones of said computing devices for execution
thereof.

15 21. The computer software product according to claim 20,
wherein said step of making respective allocations is performed
so as to minimize a completion time of said suite of test
programs.

20 22. The computer software product according to claim 20,
wherein said computer is further instructed to perform the
steps of:

coupling a new computing device to said server; and
reallocating said test programs to said computing devices
25 and said new computing device.

23. The computer software product according to claim 20,
wherein said computer is further instructed to perform the
steps of:

30 detaching one of said computing devices from said server;
and

marking unexecuted tests of said respective allocations of said one computing device as not run.

24. The computer software product according to claim 20,
5 wherein said computing devices comprise MIDP-compliant devices,
and

wherein said test programs comprise MIDlets, which are packaged in respective JAD files and JAR files, and

10 wherein allocating said test programs comprises downloading
said JAD files and said JAR files to said MIDP-compliant devices.

25. A server for testing computing devices, comprising:

15 a communication interface for coupling a plurality of said
computing devices thereto; and

a processor having instructions to access a suite of test programs for execution by said computing devices that are coupled to said server, and to distribute at least a portion of different ones of said test programs via said communication
20 interface to respective ones of said computing devices for concurrent execution thereof, said processor having further instructions to receive messages via said communication interface from said computing devices indicating completion of said test programs, and responsively to said messages, to
25 distribute remaining ones of said test programs to said computing devices for execution thereof iteratively until all of said test programs in said suite have been executed.

26. The server according to claim 25, wherein said test
30 programs are distributed as JAR files and JAD files.

27. The server according to claim 26, wherein said JAD files are constructed responsively to said messages.

28. The server according to claim 25, wherein said
5 processor has further instructions to couple a new computing device to said server; and to reallocate said test programs to said computing devices and said new computing device.

29. The server according to claim 25, wherein said
10 processor has further instructions to detach one of said computing devices from said server, and to mark unexecuted ones of said test programs that were distributed to said one computing device as not run.

15 30. The server according to claim 25, wherein said processor has further instructions to assign said different ones of said test programs in groups comprising a plurality of said test programs so as to minimize a completion time of said suite.

20
31. A server for testing computing devices, comprising:
a communication interface for coupling a plurality of said computing devices thereto; and
25 a processor having instructions to access a suite of test programs for execution by said computing devices that are coupled to said server, to assign a respective unique identifier to each of said plurality of said computing devices for use in communicating with said server, to make respective
30 allocations comprising different ones of said test programs for said computing devices, to download said allocations from said server for respective execution by said computing devices

coupled thereto, so that at least first and second computing devices among said plurality execute different first and second test programs from said suite substantially simultaneously, said processor having further instructions to receive messages
5 from said computing devices indicating completion of said execution of said test programs, each of said messages containing said respective unique identifier, and responsively to said messages to distribute remaining ones of said test programs iteratively to said computing devices for execution
10 thereof.

32. The server according to claim 31, wherein said processor has further instructions to couple a new computing device to said server; and to reallocate said test programs to
15 said computing devices and said new computing device.

33. The server according to claim 31, wherein said processor has further instructions to detach one of said computing devices from said server; and to mark unexecuted
20 tests of said respective allocations of said one computing device as not run.

34. The server according to claim 31, wherein said computing devices comprise MIDP-compliant devices, and said
25 test programs comprise MIDlets, which are packaged in respective JAD files and JAR files, and wherein said processor has further instructions to allocating said test programs by downloading said JAD files and said JAR files to said MIDP-compliant devices.

30